PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : H04L 1/00, 29/06

A1

(11) International Publication Number: WO 00/60795

(43) International Publication Date: 12 October 2000 (12.10.00)

(21) International Application Number: PCT/FI00/00233

(22) International Filing Date: 22 March 2000 (22.03.00)

(30) Priority Data:
990743 1 April 1999 (01.04.99) FI

(71) Applicant (*for all designated States except US*): NOKIA MOBILE PHONES LTD. [FI/FI]; Keilalahdentie 4, FIN-02150 Espoo (FI).

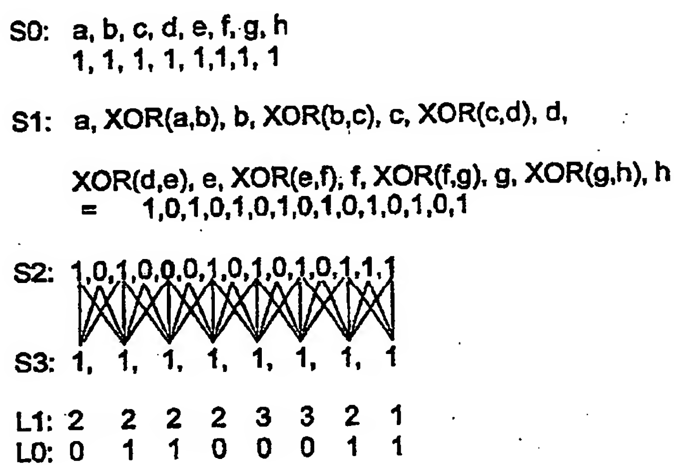
(72) Inventors; and
(75) Inventors/Applicants (*for US only*): TURUNEN, Matti [FI/FI]; Kukkuraisenkatu 4, FIN-33700 Tampere (FI). KALLIOKULJU, Juha [FI/FI]; Jokioistentie 5, FIN-37470 Vesilahti (FI). HAUMONT, Serge [FR/FI]; Riistavuorenkuja 3 B 10, FIN-00320 Helsinki (FI). LEON, David [FR/US]; 900 Meadow Creek Drive #3061, Irving, TX 75038 (US).

(74) Agent: JOHANSSON, Folke; Nokia Corporation, P.O. Box 319, FIN-00045 Nokia Group (FI).

(81) Designated States: AE, AG, AL, AM, AT, AT (Utility model), AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), DM, DZ, EE, EE (Utility model), ES, FI, FI (Utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHOD AND DEVICES FOR DIGITAL DATA TRANSFER



(57) Abstract

A data packet is subject to corruption when sent over a radio link. In case of real-time data such as video conferencing, it is preferable not to loose any data packets, because lost packets cannot anymore be used as the time for their representation has already gone by the time they could be received if sent again. To mitigate the problems caused by the limited data bandwidth it is possible to compress the header data of a packet, preferably using inter coding to maximise compression gain. This can lead to considerable data saving especially in the case of a serial link, but is a vulnerable method as a single error in data transfer can corrupt the compressed header, the payload associated with it and also all the following data packets until the inter coding is initialised again by sending one packet with an uncompressed header. The present invention concerns the use of forward error correction encoding in conjunction with header compression to protect the compressed header and optimise the radio link usage to obtain sufficient quality of received data.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

Method and devices for digital data transfer

The present invention relates to digital data transfer, especially to compression and error correction of data packets of packet mode digital data transfer in a digital mobile telephone network.

For digital data transfer it is often necessary to divide the information into packets of a certain size prior to transferring the data. Each packet comprises a certain number of bits. One part of the bits forms a payload of a packet and another part of the bits forms a header of a packet containing certain control information. In the Internet a server transferring data packets uses the headers for routing packets to their correct destinations. The protocols used to control the packets during their transfer in the Internet are TCP (Transmission Control Protocol), UDP (User Datagram Protocol) and IP (Internet Protocol). These protocols define how the data is assembled in different packets and how each packet is routed from one node to another on the way to its destination. Packet mode data transfer has certain benefits and is therefore planned to be used in the forthcoming wireless standards UMTS and GPRS (UMTS, Universal Mobile Telephone System, GPRS, General Packet Radio System). Wireless data transfer is increasing rapidly as people want to communicate and be reached independent of their location. Wireless digital networks, e.g. GSM, enable connecting a mobile telephone to another mobile telephone or to public switched telephone networks (PSTN) or other networks like ISDN (integrated services digital network). Wireless mobile telephone networks (MNW) connected to a PSTN are well known e.g. from the GSM (Global System for Mobile communications). A MNW allows a user to communicate virtually anywhere and even while travelling, e.g. in a train. It is an object of future wireless networks to support also transfer of motion video for video conferencing over radio links.

A specific Real Time Protocol (RTP) is designed for transfer of real-time data. This allows real-time communication over a packet mode connection such as video conferencing over an Internet connection and so called Internet telephone calls.

RTP may be used with UDP and IP as IP/UDP/ RTP. Video and audio data are encapsulated in RTP (Real Time Protocol) packets which are in turn encapsulated in UDP (User Datagram Protocol) and IP (Internet Protocol) datagrams. The Real-time Transfer Protocol (RTP) is designed for real-time traffic over both Intra- and Internets. RTP functions include loss detection for quality estimation and rate adaption, sequencing of data, intra- and intermedia synchronisation, source identification, and basic membership information. RTP is designed to operate on any kind of network-protocol and is therefore completely self-contained, meaning that it does not depend on any information in the lower levels of the network model. RTP is only implemented over IP/UDP today, but the protocol can be used on any type of packet-network, e.g. native ATM or ISDN. The RTP header has 12 octets, the UDP header 8 octets and the IP header 20 octets of bits. The total overhead is thus 40 octets of bits per IP datagram. Due to the fact that RTP packets often carry only small amounts of payload data, the proportion of the header information is high and a large part of the available bandwidth has to be allocated to the overhead.

The quality of both audio and video communication achievable is dependent on the maximum data transmission rate i.e. bandwidth available. Common audio CD-quality requires a bandwidth of 150 kBytes/second if no compression of the audio data i.e. payload is used. Even using the efficient MPEG 1 audio layer 3 i.e. MP3 audio compression, a bandwidth of approximately 15 kBytes/s i.e. 120 kBits/s is required. It is difficult to achieve good quality in a real-time connection capable for, say, 57,6 kbits/sec bandwidth with a wired V.90 modem or 9,6 kbits/sec bandwidth with a GSM terminal. Increasing the data transfer rate is one of the major targets in data transfer overall, and especially in wireless communication. One way to boost the data transfer rate is to expand the size of data packets and thus to reduce the overhead caused by headers. However, increasing the packet size leads to longer delays in data transfer, which is in contradiction with the general objectives of data transfer in general and especially for real-time data transfer. Also the chance of bit errors affecting the payload becomes more likely in large packets. Another method to boost transmission utilises compression of data packets using algebraic algorithms to encode the data to be transferred into a

reduced number of bits. This yields faster communication as the number of bits for transmission decreases. The payload may be compressed using payload type specific compression algorithms and some header specific form may be applied to the headers of the packets. In these methods the header and the payload are compressed separately, because the header has in some cases such characteristics that allow more powerful compression when the header is compressed alone. It is also easier to decompress the header again for further routing when the payload and header are compressed independently.

In the Internet, sequentially sent data packets may be routed via different nodes. This enables use of plurality of parallel data connections between the sender and the recipient. Fixed packet-switched networks have a high data bandwidth. On the other hand, data packets may not arrive at their destination in the same order in which they were sent, because sequential packets may be routed via different nodes and the data bandwidth may differ from one route to another. It is therefore necessary to include complete information concerning the destination in each data packet. For these reasons, the header must contain a certain amount of information, but the size of the headers is usually not a problem thanks to the high data bandwidth. However, problems arise if packets have to be transmitted on a serial link with a low bandwidth. This is often the case when a home user accesses access points of the network, e.g. using a personal computer. Such access links are typically telephone line modems, which have been used for several years, and more recently wireless connections. In the particular case of GSM, data connections, HSCSD (High Speed Circuit Switched Data) and GPRS are all examples of serial links. Due to their serial nature, these kind of links, as opposed to core packet-switched networks, have the property that packets are received in the order they were sent. There is no way a packet may overtake its predecessor, because each packet is transferred in sequence. This sequential reception of data packets and the properties of packets in the common TCP/IP protocol enable compression of the header data very efficiently using header compression protocols.

The publications: "Compressing TCP/IP Headers for Low-Speed Serial Links" by V. Jacobson, February 1990 and "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links" by S. Casner and V. Jacobson, July 27, 1998 describe methods for compressing headers of data packets. The latter of the two publications claims that it is even possible to achieve compression from 40 bytes of the original header down to 3 bytes. The idea behind efficient compression of headers is based on the characteristics of TCP/IP headers, which have certain usually constant behaviour. Firstly, one part of the header remains constant during the entire connection. Secondly, another part of the header consistently changes from one packet to another, allowing inter coding. The change can be consistent, but even if it is not, encoding the difference between specific parts of headers of sequential packets requires less information than sending the absolute values of these parts. Encoding data utilising constantly changing behaviour of the data is known as inter coding. In the publication of Casner and Jacobson it is however mentioned that the compression scheme presented performs best on local links with low round-trip-time, which is not the case in usual digital mobile networks.

In effect, serial transmission and reception of packets enables the efficient use of inter coding of a series of headers. To understand what inter coding is, consider the simple example of differential (or delta) encoding. Instead of transmitting a series of numbers a, b, c, d, e, etc. the sequence a´=a, b´=b-a, c´=c-b,.. is transmitted. Encoding the difference requires lower resolution i.e. data word length than encoding the absolute values, thus saving again some bits. However, the weakness of such a scheme is that if one of the inter coded headers is lost, the packet associated with it and all following packets will be lost. To recover from the damage of the header there are two methods: 1) Sending periodically a data packet without header compression to refresh the series of headers i.e. to re-initiate inter coding and 2) refreshing the series of headers on demand when a loss is detected.

Periodical refresh of the series of headers wastes some of the compression benefit by forcing packets to be sent uncompressed even when there is no need for that. Refresh on demand is based on an acknowledgement mechanism. The

recipient indicates whether the header of the received data packet is corrupted. Use of acknowledgements may be satisfactory for modem links but is not sufficiently suited for wireless links for the following reasons:

- The error rate is typically several orders of magnitude higher in wireless transmission than in fixed networks. Motion of the mobile station increases the likelihood of errors due to the changes in radio coverage and Doppler shift. The Doppler shift causes a change in the radio frequency in the reception of radio data and causes data corruption when the movement is rapid, e.g. when the receiver is used in a moving train.
- The packet round-trip delay is considerable because of interleaving, even up to several hundreds of ms. Interleaving in this context means that prior to transmitting the data, channel encoding may spread the data bits of single data packet between two or more blocks of packets. Thus, gathering the complete data for an entire packet may require reception of two or more blocks, which are combined to produce the original packets in their correct order. This adds a significant delay.

Correcting for packet loss using retransmissions causes a delay in data transfer. Delays are most harmful for two-way communication. In a unidirectional audio connection such as listening to sound clips or messages from a remote answering machine, the receiver can adapt to the variability in the delay. Delays of even a few hundreds of milliseconds can interfere with the flow of conversation in a two-way link. If header compression by intercoding is used, there is a risk that an entire sequence of packets may be lost due to corruption of a single header (as explained earlier). Therefore header compression is barely suitable in real-time communication such as video conferencing over a radio interface. Radio links are susceptible to radio interference and can have a bit error rate up to 10^{-4} whereas corresponding figure for PSTN is typically less than 10^{-7}. An error in any one of the bytes of a compressed header may cause the entire packet to be lost, because without the header the packet cannot be routed to its destination. Such vulnerability is a major disadvantage and restricts the use of header compression for e.g. real-time motion video and/or voice transfer. When transferring stored files, retransmission of failed packets is possible. In real-time communication each packet is used as soon as possible after reception to minimise the data transfer

delay. If one packet is missed, it cannot be used any more after reception of the next packet because the time has lapsed when that information was to be represented for the recipient. On the other hand off-line transmission can utilise large data packets as the cycle time is not critical and this inherently reduces the overhead due to the header data. Thus, in the case of file transfer, it is not as necessary as in real-time communication to reduce the size of the header. One typical example of an error-prone communication channel is a radio link between a mobile communication station like a GSM-handset and a base station.

For the reasons given above, in the case of the mobile telecommunication, inter coding of headers is currently considered to be inappropriate for real-time operation like RTP even if either one of the two methods to re-initiate inter coding is used.

However, it would be nevertheless beneficial if header compression could be applied in real-time communication, particularly as the amount of bandwidth required to transmit uncompressed headers is significant compared with that required to transmit the packet payload (data). In their publication, Casner and Jacobson addressed the same problem, giving figures of 20 bytes for the payload and 12 bytes for an RTP header. Referring to those figures the overhead caused by the header is 12 bytes/20 bytes = 60 %. It is thus obvious, that there is a great desire to enable use of header compression also over data links subject to corruption.

A new method and devices for error tolerant data packet transfer have now been invented. According to the present invention, the header of a packet is first compressed and thereafter protected to some extent against bit errors using Forward Error Correction (FEC). This enlarges the size of the header by some amount due to the introduction of forward error correction information, but enables at least recovery of headers corrupted by one bit error during transfer. It should be noted, that although any form of forward error correction involving adding redundancy has previously been avoided in order to allocate the maximum proportion of the bandwidth to transfer the actual payload, using forward error

correction to protect compressed headers against bit errors actually provides an overall improvement in data transfer rate. The method can be implemented, for example, in telecommunication devices intended for use in packet switched radio networks like GPRS.

The method introduced by this invention utilises header compression followed by forward error correction (FEC) coding of the compressed header. Here the concept of "forward error correction" coding means adding redundancy bits to enable recovery of corrupted data using the redundant bits without necessarily having to retransmit data. Use of FEC to protect headers enables recovery of a compressed header having a certain number of bit errors. The maximum number of recoverable errors in the header depends on the FEC algorithm used and the number of redundant bits added to the compressed header. Even though use of FEC adds redundancy, data transfer efficiency over error prone radio links improves more than FEC redundancy impairs it and thus the practical transfer rate improves. This is especially advantageous in real-time communication over a error-prone communication channel such as a mobile communications system, but in general this is useful when the bandwidth of a communication channel is limited, as usual in radio communication, or over a fixed telephone/modem link. The present invention is even more advantageous for low bit rate real-time communication over an error-prone communication channel. With low-speed and error-prone communications, the influence of errors is most considerable when they occur in any of the bits of a compressed header of a data packet.

Still advantageous to the invention, if one bit error affects the packet, instead of loosing all the rest of the stream until the next refresh, it will be possible:

- To first of all detect if the bit error has affected the header or the payload (e.g. audio, video or data). If it is only the payload, it can be decided to either discard the payload or apply some forward error correction. In any case, the packets following thereafter can be used since the inter coded header is intact.
- If the compressed header is itself damaged, the added forward error correction bits can be used to attempt recovery of the original header, and if the recovery succeeds, the data stream can be used, lowering the probability that a refresh (re-

transmission of an entire packet without compression) is needed. Efficiently protecting the bits of a compressed header is reasonable since after compression the header typically consists of only of a few bytes, and although certain amount of redundancy is introduced this may make it unnecessary to retransmit entire data packets. This may significantly reduce the amount of data to be retransmitted and will result in a significant improvement in a two-way communication by reducing the number of disturbing pauses in the video/audio reception.

According to a first aspect of the invention a method is provided for transmitting and further receiving a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said method further comprises the steps of:

compressing said group of bits;

modifying said header by forward error correction encoding the compressed group of bits;

sending said data packet over said transmission path;

receiving said data packet;

forward error correction decoding the compressed group of bits; and

decompressing the compressed group of bits to regenerate said header.

According to a second aspect of the invention a method is provided for transmitting a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said method comprises the steps of:

compressing said group of bits,

modifying said header by forward error correction encoding the compressed group of bits; and

sending said data packet over said transmission path after the error correction encoding.

According to a third aspect of the invention a method is provided for receiving a compressed digital data packet comprising a header and a payload, said header comprising a group of bits, characterised in

receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

decoding said forward error correction encoded group of bits; and

decompressing said compressed group of bits.

According to a fourth aspect of the invention digital radio telephone network is provided for transferring a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said network comprises:

means for compressing said header by compressing said group of bits;

means for forward error correction encoding of the compressed group of bits for protecting said header against a bit error in said compressed group of bits;

means for sending said data packet over said transmission path;

means for receiving said data packet;

means for recovering said encoded group of bits by decoding it; and

means for decompressing said header.

According to a fifth aspect of the invention a digital data transmitter is provided for transmitting a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said transmitter comprises:

means for compressing said group;

means for error correction encoding of said group using a forward error correction method; and

means for sending said data packet over said transmission path.

According to a sixth aspect of the invention digital data receiver is provided for receiving a digital data packet sent over a digital transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said receiver comprises:

means for receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

means for forward error correction decoding said group of bits; and

means for decompressing said group of bits.

According to a seventh aspect of the invention a mobile station is provided, comprising a digital data transmitter for transmitting a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said mobile station comprises:

means for compressing said group;

means for error correction encoding of said group using a forward error correction method; and

means for sending said data packet over said transmission path.

According to an eighth aspect of the invention a mobile station is provided, comprising a digital data receiver for receiving a digital data packet sent over a digital transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, characterised in that said mobile station comprises:

means for receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

means for means for decoding said forward error corrected encoded group of bits decoding said forward error corrected group of bits; and

means for decompressing said group of bits.

According to a ninth aspect of the invention a packet data network element is provided for error resilient communications, comprising means for receiving a digital data packet sent over a digital transmission path, said packet comprising a payload and a header comprising a group of bits, characterised in that said element comprises:

means for receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

means for means for decoding said forward error corrected encoded group of bits decoding said forward error corrected group of bits; and

means for decompressing said group of bits.

According to a tenth aspect of the invention a packet data network element is provided comprising means for transmitting a digital data packet over a transmission path, said packet comprising a payload and a header comprising a group of bits, characterised in that said element comprises:

means for compressing said group;

means for error correction encoding of said group using a forward error correction method; and

means for sending said data packet over said transmission path.

The invention will now be described by way of example with reference to the drawings in which:

Figure 1 shows telecommunication network connections in the packet switched GPRS network known from prior art;

Figure 2 is a diagram of simplified data flow over a packet switched radio link subject to data corruption known from prior art;

Figure 3 is a diagram of a digital data packet known from prior art;

Figure 4 is a block diagram of the parts of a mobile station according to an embodiment of the present invention;

Figure 5 is a diagram of a compressed header known from prior art;

Figure 6 is an example scheme of locations of forward error correction bits in relation to the bits they are intended to protect according to an embodiment of the invention;

Figure 7 is a diagram of an example of construction of a compressed and then FEC encoded header, damaging of a compressed and FEC encoded header and further reconstruction of a damaged header according to an embodiment of the invention;

Figure 8 is a diagram of the bytes of a compressed and forward error encoded header in accordance with an embodiment of the invention; and

Figure 9 Is a diagram of a compressed and forward error encoded header in accordance with figure 8.

Figure 1 presents telecommunication network connections in a typical packet switched GPRS network. The main element of the GPRS network infrastructure is a GPRS support node, a so called GSN. It is a mobility router that effects the connection and co-operation with the different data networks, e.g. with the public packet data network PSPDN (Public Switched Packet Data Network) through connection Gi or with another operator's GPRS network through connection Gp, mobility management by GPRS registers through connection Gr and relaying of data packets to mobile stations independent of their locations. Physically, a GPRS Support Node GSN can be integrated into an MSC (Mobile Switching Center) or it can be a separate network element based in the architecture of data network routers. User data passes directly through connection Gb between the support node GSN and the Base Station System BSS, which consists of Base Stations BTS and Base Station Controllers BSC, but between the support node GSN and the Mobile Switching Center MSC there is the signalling connection Gs. The solid lines between blocks represent data traffic and the dashed lines represent signalling. Physically, the data can pass transparently through the Mobile Switching Center MSC. The radio interface between the Base Station BTS and the mobile station MS is marked by reference Um. The references Abis and A represent the interface between the Base Station BTS and the Base Station Controller BSC and the interface between the Base Station Controller BSC and the Mobile Switching Center MSC, which is a signalling connection. The reference Gn represents the connection between different support nodes within the same operator's network. The support nodes are usually classified as gateway support nodes GGSN (Gateway GSN) and serving or home support nodes SGSN (Serving GSN) as presented in figure 1.

Figure 2 shows a simplified diagram of data flow over a packet switched radio link subject to data corruption, known from prior art. For simplicity it is assumed that interleaving is not used in the radio connection between the mobile station MS and the telecommunication network NW. Thus each packet is assumed to be transmitted in its entirety in a separate frame. This is in contrast with the normal situation, for example in real GPRS data connections, where it is customary to interleave the bits of each data packet, dividing them into different frames.

Interleaving as such is well known from prior art to those normally skilled in the art. For description of the invention it is irrelevant whether each packet is sent as such or interleaved into different packets. Even when interleaved, the header, as well as the payload of a packet, may be corrupted during the data transfer and the corruption has the same impact as described earlier.

As shown in figure 2, a mobile station MS sends a radio signal formed by a stream of RTP data packets P1 to P5 over a radio link to a telecommunication network NW. The network normally comprises a terminal receiving the data (not shown). The data stream comprises a number of sequential data packets, each comprising a certain number of bits. Some of the bits are required for controlling the routing of each packet, and other bits carry the actual information (payload) to be transferred. In the example shown in figure 2, it is assumed that during the transfer there is a short period of interference affecting the data of packets P2 and P3. It is now supposed, that the errors caused in packet 2 affect only its payload, and the header of packet P3 becomes corrupted. The corruption of payload impairs the quality of the data received, but audio and/or video coding algorithms, e.g. error detection, correction and concealment algorithms, can be applied to reduce the impact of errors in the payload. However, corruption of header of packet 3 will render the entire packet useless as the control information for routing the packet to its destination is lost or becomes invalid. Furthermore, if an inter coding method is used to compress the headers, all following packets will also become unusable until the MS receives an acknowledgement informing it of a failure to receive the packets P3 to P5, or until the MS sends an uncompressed packet, that is a packet without inter coding, as a periodical refresh of inter coding. The number of missed packets depends on the cycle (round-trip) time i.e. time lapsed between transmitting a packet from the mobile and receiving the acknowledgement that a certain packet has not been correctly received. A typical value of the cycle time is 330 milliseconds. If packets are sent every 30 ms (a typical value) the average number of lost packets due to an error in packet header equals to 330 ms/30 ms = 11 packets. Or, expressed in time, there will be an interruption in the reception of data (e.g. audio and/or video data) for a time at least equal to the cycle time i.e. 330 ms.

Figure 3 illustrates a digital data packet P1 known from prior art. The packet comprises a header 11 and a payload 12. The header comprises an inter coded group of bits (30) forming at least part of the header data.

Figure 4 is a block diagram of the parts of a mobile station MS which are useful for describing the invention according to an embodiment of the present invention. The mobile station comprises a central processing unit CPU controlling the mobile station. The instructions for the CPU are stored in the memory MEM. For reception of data packets there is a receiver module RX connected to aerial AER via duplex filter DPX. For transmitting of data packets, the mobile station has a transmitter module TX connected to the aerial AER via the duplex filter. Additionally, there may be a user interface UI for user interaction and a data interface IF for connecting to an auxiliary device, e.g. a laptop computer. The CPU is arranged to organise the data to be sent into data packets and to form headers for the data packets. The CPU is also arranged to separate the useful data i.e. the payload from data packets received by the RX. Alternatively, processing of data packets may be performed by specific hardwired circuitry, a specific or shared digital signal processor (DSP) or any other equipment for data processing known from prior art.

Figure 5 illustrates a compressed header 31 consisting of four bytes, PB1, PB2, PB3 and PB4. The eight bit positions of the first byte PB1 of the compressed header 31 have reference numerals 1 to 8.

In a radio link there is a high likelihood of burst-type errors and therefore it should be appreciated that each forward error correction bit may be located on purpose far away from the bits of the header it is aimed to protect. One scheme would be to place the forward error correction bits among the header data at a constant distance from the bits they are computed from. A preferred option is to calculate the forward error correction bits from the bits of the header located far away from each other. This improves the chance that only one of the protected header bits is corrupted when they are attempted to recover by a common forward error

correction data. Although there are number of ways to implement this, one example is given in figure 6.

In figure 6 a compressed header comprising 4 bytes i.e. 32 bits of header information is arranged in four rows of bits. In this example a parity bit is computed for every two bits of header data. In order to illustrate how the forward error correction bits relate to the other bits of the 4 bytes of this example, each bit position corresponding to a single bit is labelled sequentially with a letter from a to z and then from α to η. The total size of the header to be transferred is 6 bytes as the compressed header comprising 4 bytes is increased by 2 bytes for forward error correction. Starting from the second row, after each bit there is an forward error correction bit labelled with a combination of two characters representing two different bits. Each forward error correction bit is computed from a bit of the preceding byte and a bit of the following byte. Thus bit j is followed by forward error correction bit $\genfrac{}{}{0pt}{}{a}{r}$, which is computed from the bits a and r. Respectively, after bit k there is bit $\genfrac{}{}{0pt}{}{b}{s}$ and so on. The bits a to i form byte 1, bits from j to $\genfrac{}{}{0pt}{}{d}{u}$ form byte 2, bits from n to $\genfrac{}{}{0pt}{}{i}{y}$ form byte 3, bits r to $\genfrac{}{}{0pt}{}{m}{\chi}$ form byte 4, bits from v to $\genfrac{}{}{0pt}{}{q}{h}$ form byte 5 and bits from z to η form byte 6. When each of the six bytes are sent in sequence, the possible occurrence of a burst of errors could corrupt multiple bits in sequence and yet it would be possible to recover the contents of the header. For example, if the entire byte 2 would be corrupt or lost, that would mean corruption of four bits of the header and four forward error correction bits. As none of the corrupted forward error correction bits would refer to a corrupted header bit, losing the forward error correction bits of byte 2 would not harm recovery of the header bits of the same byte 2. Instead, the forward error correction bits to be used would be interleaved within byte 4 and thus could be used to recover the header bits j to m of the corrupted byte 2. The function for computation of forward error correction bit values can be for example an XOR function.

To indicate and/or recover from errors occurring in the payload, error detection and/or forward error correction encoding can also be focused to that part of a data packet.

Figure 7 shows a diagram of an example of encoding of a header, damage caused by data transfer and further reconstruction of a damaged compressed header according to an embodiment of the invention. The bit positions of consecutive bits of a byte to be forward error correction encoded are referred to by letters a,b,c,d,e,f,g,h. The forward error correction encoding scheme is as follows: a, XOR(a,b), b, XOR(b,c), c, ... The scheme is illustrated with 8 bits i.e. one byte. The original sequence is S0. Each bit can have any of the two values, 0 and 1, but to simplify they are all set to 1. The encoded sequence is shown as sequence S1. Then an imaginary error has changed the value of the third encoded and the second last bit resulting in the sequence S2. Next the encoded bits of the header are FEC decoded. When the corrupted sequence S2 is received, the sequence S3 is computed using the added redundancy. To find out which bits are corrupted a likelihood analysis is performed.

Each of the bits has an equal probability to be inverted. Thus bit errors are equally likely to affect the compressed data content of the header or the parity bits included in the header. In order to determine whether a bit error has occurred during data transmission, it is necessary to examine the received bit stream. In this example between every two data bits of header there is a parity bit computed from these data bits. With the exception of the first and last data bits, the value of each data bit can be derived from three independent and equally reliable sources. For a certain data bit (*e*):

- The received data bit *e* as such has a certain first value.
- For certain data bit (*e*) the previous data bit (*d*) and the parity bit $\binom{d}{e}$ between that certain data bit (*e*) and the previous data bit (*d*) indicate a second value for that certain data bit acting as the second source.
- The data bit (*f*) after the certain data bit (*e*) and the parity bit $\binom{e}{f}$ indicate the third value for that certain data bit acting as the third source.

If all these three sources indicate to the same value for a certain data bit, it is probable that that data bit is correctly received (first value). In case that two of these bits support the first value it is also more likely that the first value is correct than incorrect. However, if both the second and third values differ from the first value, it is more probable that the first value is incorrect and it should be inverted to result in a value that is probably correct. For the first and last data bit this does not apply, because there are only two possible indications to decide on the correct value of the data bit. For these data bits it impossible to decide whether their values are correct as it is unsure whether the data bit in one of the ends of the sequence or the parity bit next to it is incorrect. Both of these bits have, again, an equal possibility to be incorrect. To protect all the data bits, the sequence of data bits could be extended by one constant bit (e.g. 1) from each of the two ends.

This is further illustrated using an example with two bit errors, one of them in a data bit and another one in a parity bit. When the sequence S2 is received, the following sequence is started:

1) An array is created to hold two likelihood counters for each data bit. The first counter L0 is intended to count observations indicating that the corresponding bit should be zero. The second counter L1 counts observations indicating that the corresponding bit should be one. Both counters are first set to zero.

2) The bit value of the first bit i.e. the first data bit *a* is the first observation. If this bit is a 0, then counter L0 for the first data bit is incremented by 1, otherwise counter L1 is incremented by 1.

3) Next the two following bits i.e the first parity bit $\genfrac{}{}{0pt}{}{a}{b}$ and the second data bit *b* are read. Then the first data bit a and second data bit b are compared by an XOR operation and the result is further compared to received first parity bit $\genfrac{}{}{0pt}{}{a}{b}$. If these match then the counter corresponding to the value of *a* is incremented by 1, if not, then the counter corresponding to the other possible value of *a* is incremented by 1. In both cases it is still the counter corresponding to the first data bit that is used.

4) The next step is to compute values for the counters L0 and L1 corresponding to the second data bit. This time there are three different indications that can each hint to the correct value of the second data bit *b*. First of these indications is the

value of bit *b*. As in step 2, depending on the value of bit *b*, either the counter L0 or L1 is incremented by 1 i.e. if *b*=0 then L0 is incremented by 1, otherwise L1 is incremented.

5) The second indication is the equality of the parity bit $\frac{a}{b}$ and parity computed from bits *a* and *b*. Again, if these match, then the counter L0 or L1 corresponding to the value of bit *b* is incremented by 1. Otherwise, the other counter is incremented.

6) The third indication is the equality of the parity bit $\frac{b}{c}$ and parity computed from bits b and c. Again, if these match, then the counter L0 or L1 corresponding to the value of bit *b* is incremented by 1.

7) Each data bit is examined according to steps 2-6. This procedure yields two arrays of likelihoods, L0 and L1.

8) Each data bit having two or three similar indications (L0 or L1 is 3) is accepted as such.

9) Each data bit having two contraindications, i.e. two indications against the received value, is inverted.

10) The first and last data bit are held unchanged. If there is a contraindication the value of the data bit may be considered insecure.

Figure 8 is a diagram of the bytes of a compressed and forward error encoded header 41 in accordance with an embodiment of the invention. The header comprises a sequence of bits formed by three bytes of header data and also three bytes FEC1, FEC2 and FEC3 of error correction data.

Figure 9 is a diagram of a compressed and forward error encoded header in accordance with figure 8. To illustrate the layout of the bits within the header, the contents of each header data byte is shown as bits marked with letters running from a to x. Each bit of FEC bytes is also marked with letters corresponding to the header data bits from which it is calculated. In the header, the bits are organised in a linear series starting from bit a until the last FEC bit marked with x. The number of FEC bits is 23 in this error correction scheme as a parity bit is formed for each two adjacent bits. To fill the last bit position of the third FEC byte a header data bit can be copied as such, preferably one of the first and last data bits to strengthen

its protection against bit errors. Alternatively that last bit position could be padded for example by the first bit of the payload.

According to the invention the telecommunication network also comprises means for decompressing the compressed and FEC encoded headers into a standard form to allow their normal use. This can be implemented in any appropriate part of the network, e.g. at the MSC or BSC, a CPU and memory or specific hardware alike described earlier referring to figure 4.

It will be appreciated by the skilled person that modification may be made to the above described embodiment without departing from the scope of the present invention. For example, more sophisticated forward error correction methods, similar to that described in the US-patent 5870412, may be used instead of the simple XOR-method described above, to obtain better error correction capability with same amount of redundancy.

Claims

1. Method for transmitting and further receiving a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, **characterised** in that said method further comprises the steps of:

compressing said group of bits;

modifying said header by forward error correction (FEC) encoding the compressed group of bits;

sending said data packet over said transmission path;

receiving said data packet;

FEC decoding the compressed group of bits; and

decompressing the compressed group of bits to regenerate said header.

2. Method for transmitting a digital data packet over a transmission path subject to data corruption, said packet comprising a payload and a header comprising a group of bits, **characterised** in that said method comprises the steps of:

compressing said group of bits,

modifying said header by forward error correction (FEC) encoding the compressed group of bits; and

sending said data packet over said transmission path after the error correction encoding.

3. Method for receiving a compressed digital data packet comprising a header and a payload, said header comprising a group of bits, **characterised** in that said method comprises the steps of:

receiving a data packet comprising said header where said group of bits is compressed and forward error correction (FEC) encoded;

decoding said forward error correction encoded group of bits; and

decompressing said compressed group of bits.

4. Method according to claim 1 or 2, **characterised** in that said transmission path is a channel of a mobile telecommunications path and said data corruption is caused by radio interference.

5. Method according to claim 1 or 2, **characterised** in that said step of compressing said group of bits comprises inter coding said group of bits.

6. Method according to claim 5, **characterised** in that said step of decompressing said compressed group of bits comprises decoding the inter coded group of bits.

7. Method according to any one of preceding claims, **characterised** in that said payload comprises real-time communication data.

8. Digital radio telephone network (MNW) for transferring a digital data packet (P1) over a transmission path (Um) subject to data corruption, said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said network comprises:

means for compressing (CPU,MEM) said header by compressing said group of bits;

means (CPU) for forward error correction (FEC) encoding of said compressed group of bits (30) for protecting said header against a bit error in said compressed group of bits;

means (TX,DPX,AER) for sending said data packet over said transmission path;

means (RX,DPX,AER) for receiving said data packet;

means (CPU,MEM) for decoding said FEC encoded group of bits; and

means (CPU,MEM) for decompressing said header.

9. Digital radio telephone network according to claim 8, **characterised** in that said means for compressing said header being inter coding means.

10. Digital radio telephone network according to claim 8 or 9, **characterised** in that said transmission path is a channel of a mobile telecommunications path and said data corruption is caused by radio interference.

11. Digital radio telephone network according to any of claims 8 to 10, **characterised** in that said network comprises:

means for detecting (CPU,MEM) corruption of said group in connection with reception of the data packet, and

said means (CPU,MEM) for decoding being arranged to recover said group of bits in case said group of bits is corrupted.

12. Digital radio telephone network according to any of claims 8 to 11, **characterised** in that said payload (12) comprises real-time communication data.

13. Digital data transmitter (TX) for transmitting a digital data packet (P1) over a transmission path (Um) subject to data corruption, said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said transmitter comprises:

means (CPU,MEM) for compressing said group;

means (CPU,MEM) for error correction encoding of said group (30) using a forward error correction (FEC) method; and

means (TX,DPX,AER) for sending said data packet over said transmission path.

14. A digital data transmitter according to claim 13, **characterised** in that said means for compressing said group of bits is inter coding means.

15. A digital data transmitter according to claim 13 or 14, **characterised** in that said transmission path is a channel of a mobile telecommunications path and said data corruption is caused by radio interference.

16. A digital data transmitter according to any of claims 13 to 15, **characterised** in that said payload (12) comprises real-time communication data.

17. Digital data receiver (RX) for receiving a digital data packet (P1) sent over a digital transmission path (Um) subject to data corruption, said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said receiver comprises:

means (RX,DPX,AER,CPU,MEM) for receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

means (CPU,MEM) for means for decoding said forward error corrected encoded group of bits decoding said forward error corrected group of bits; and

means (CPU,MEM) for decompressing said group of bits.

18. Digital data receiver (RX) according to claim 17, **characterised** in that said header of the received data packet is compressed by inter coding.

19. Digital data receiver (RX) according to claim 17 or 18, **characterised** in that said transmission path is a channel of a mobile telecommunications path and said data corruption is caused by radio interference.

20. Digital data receiver according to any of claims 17 to 19, **characterised** in that said receiver comprises:

means for detecting (CPU,MEM) corruption of said group of bits in connection with reception of the data packet, and

said means for decoding said group of bits being arranged for recovering said group of bits by said decoding in case said group is corrupted.

21. Digital data receiver according to any of claims 17 to 20, **characterised** in that said payload (12) comprises real-time communication data.

22. Mobile station (MS) comprising a digital data transmitter (TX) for transmitting a digital data packet (P1) over a transmission path (Um) subject to data corruption, said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said mobile station comprises:

means (CPU,MEM) for compressing said group;

means (CPU,MEM) for error correction encoding of said group (30) using a forward error correction (FEC) method; and

means (TX,DPX,AER) for sending said data packet over said transmission path.

23. A mobile station (MS) according to claim 22, **characterised** in that said means for compressing said group of bits is inter coding means.

24. A mobile station (MS) according to claim 22 or 23, **characterised** in that the transmission path is a channel of a mobile telecommunications path and said data corruption is caused by radio interference.

25. A mobile station (MS) according to any of claims 22 to 24, **characterised** in that the payload (12) comprises real-time communication data.

26. Mobile station (MS) comprising a digital data receiver (RX) for receiving a digital data packet (P1) sent over a digital transmission path (Um) subject to data corruption, said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said mobile station comprises:

means (RX,DPX,AER,CPU,MEM) for receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

means (CPU,MEM) for means for decoding said forward error corrected

encoded group of bits decoding said forward error corrected group of bits; and

means (CPU,MEM) for decompressing said group of bits.

27. Mobile station (MS) according to claim 26, **characterised** in that said header of the received data packet is compressed by inter coding.

28. Mobile station (MS) according to claim 26 or 27, **characterised** in that said transmission path is a channel of a mobile telecommunications path and said data corruption is caused by radio interference.

29. Mobile station (MS) according to any of claims 26 to 28, **characterised** in that said receiver comprises:

means for detecting (CPU,MEM) corruption of said group of bits on or after reception of the data packet, and

said means for decoding said group of bits being arranged for recovering said group of bits by said decoding in case said group is corrupted.

30. Mobile station (MS) according to any of claims 26 to 29, **characterised** in that said payload (12) comprises real-time communication data.

31. Packet data network element for error resilient communications, comprising means (RX) for receiving a digital data packet (P1) sent over a digital transmission path (Abis,A, Gb,Gs,Gl), said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said element comprises:

means for receiving a data packet comprising a header where said group of bits is compressed and forward error correction encoded;

means for decoding said forward error corrected encoded group of bits decoding said forward error corrected group of bits; and

means (CPU,MEM) for decompressing said group of bits.

32. Packet data network element according to claim 31, **characterised** in that said element is selected from a group consisting of: Gateway GPRS support node, Serving GPRS support node, Mobile switching centre, a router and a firewall.

33. Packet data network element according to claim 31 or 32, **characterised** in that said header of the received data packet is compressed by inter coding.

34. Packet data network element according to any of claims 31 to 33, **characterised** in that said element comprises:

means for detecting (CPU,MEM) corruption of said group of bits on or after reception of the data packet, and

said means for decoding said group of bits being arranged for recovering said group of bits by said decoding in case said group is corrupted.

35. Packet data network element according to any of claims 31 to 34, **characterised** in that payload (12) comprises real-time communication data.

36. Packet data network element for error resilient communications, comprising means for transmitting a digital data packet (P1) over a transmission path (Abis,A,Gb,Gs,Gi), said packet comprising a payload (12) and a header (11) comprising a group of bits (30), **characterised** in that said element comprises:

means (CPU,MEM) for compressing said group;

means (CPU,MEM) for error correction encoding of said group (30) using a forward error correction (FEC) method; and

means for sending said data packet over said transmission path.

37. A packet data network element according to claim 36, **characterised** in that said means for compressing said group of bits is inter coding means.

38. A packet data network element according to claim 36 or 37, **characterised** in that the payload (12) comprises real-time communication data.

39. Packet data network element according to any of claims 36 to 38, **characterised** in that said element is selected from a group consisting of: Gateway GPRS support node (GGSN), Serving GPRS support node (SGSN), Mobile switching centre (MSC), a router and a firewall.

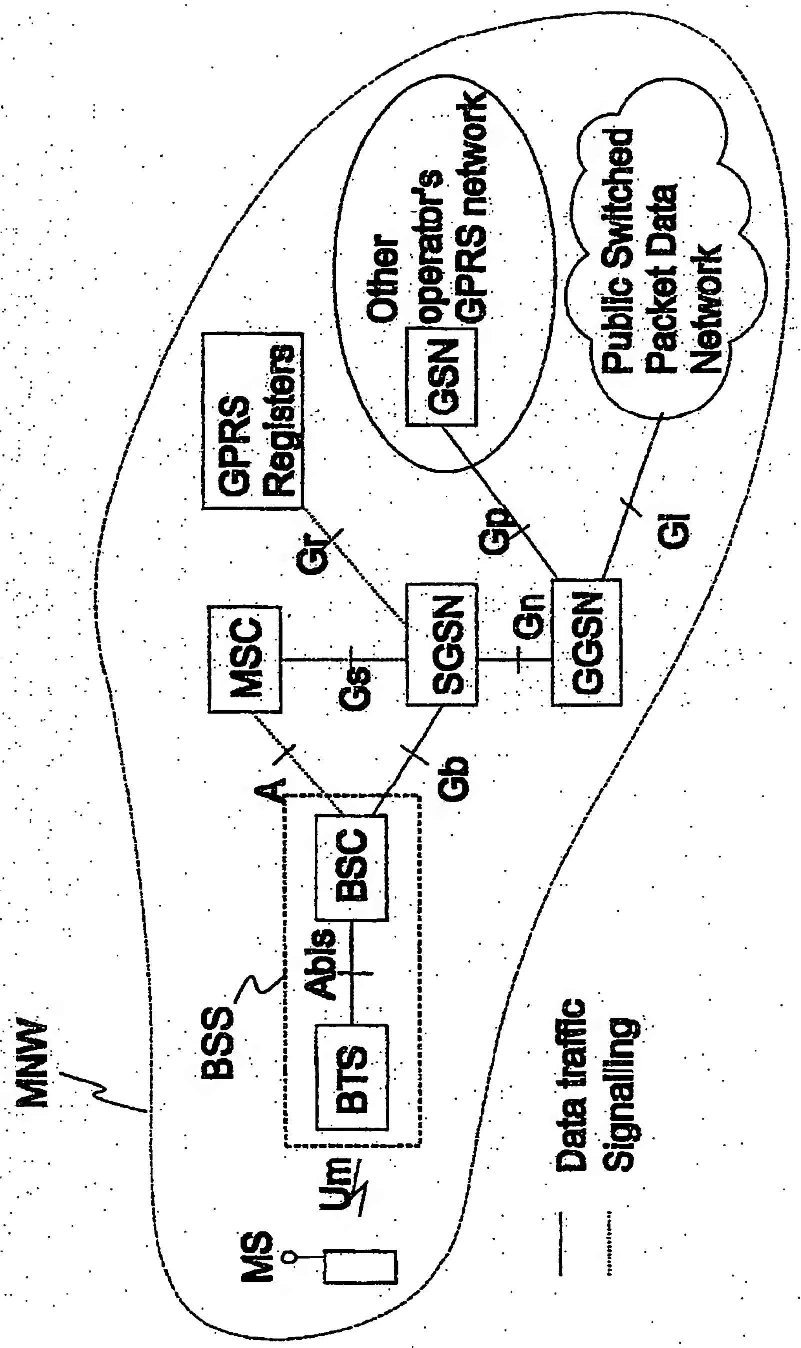

MNW
BSS
BTS
Abis
BSC
MS
Um
A
Gb
MSC
Gs
SGSN
Gn
GGSN
Gr
GPRS Registers
Gp
GSN
Other operator's GPRS network
Gi
Public Switched Packet Data Network
Data traffic
Signalling

Fig. 1

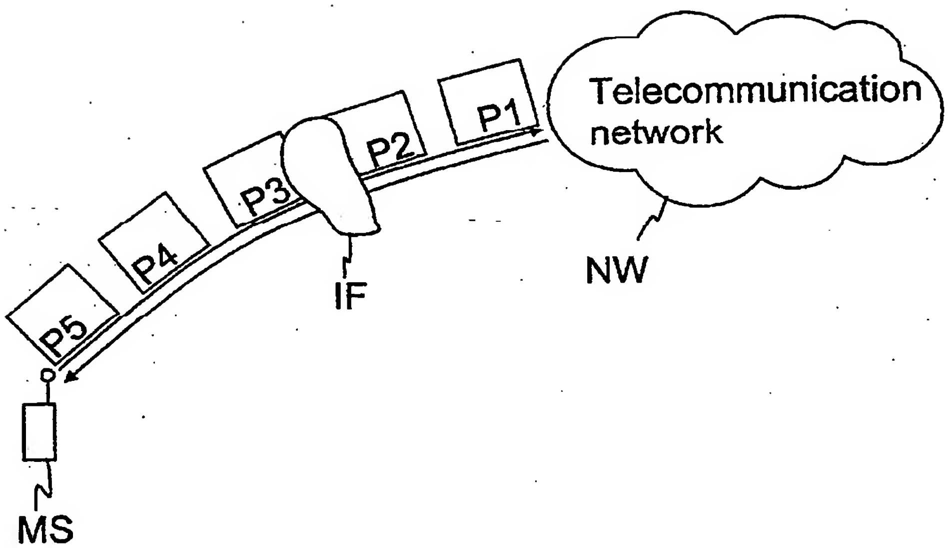

Telecommunication network
P1
P2
P3
P4
P5
IF
NW
MS

Fig. 2

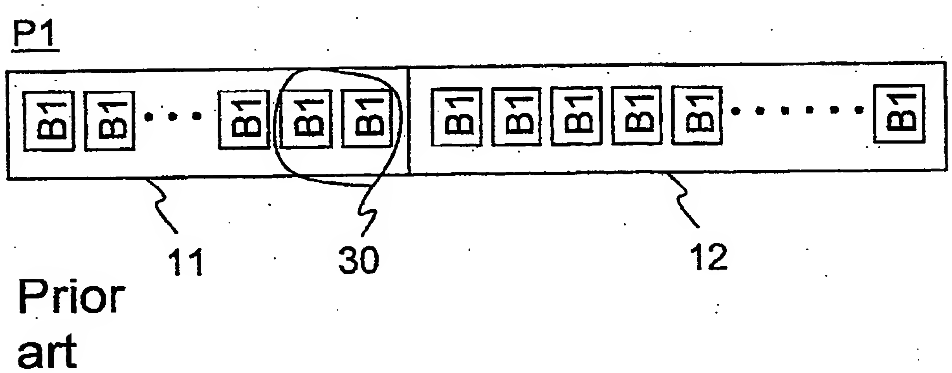

P1
B1
B1
B1
B1
B1
B1
B1
B1
B1
B1
B1
11
30
12
Prior art

Fig. 3

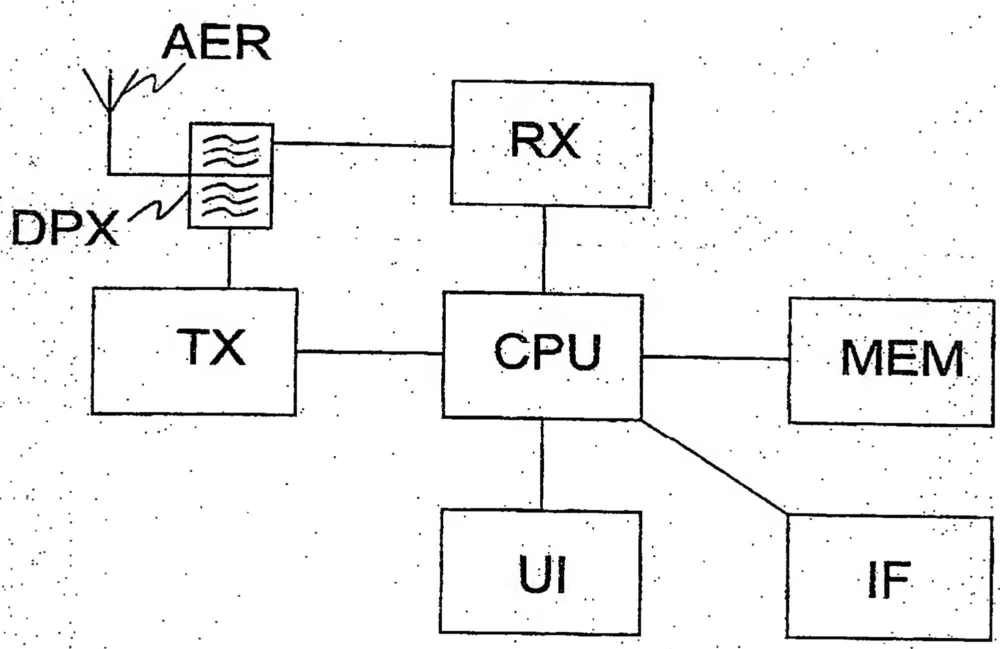

AER
DPX
RX
TX
CPU
MEM
UI
IF

Fig. 4

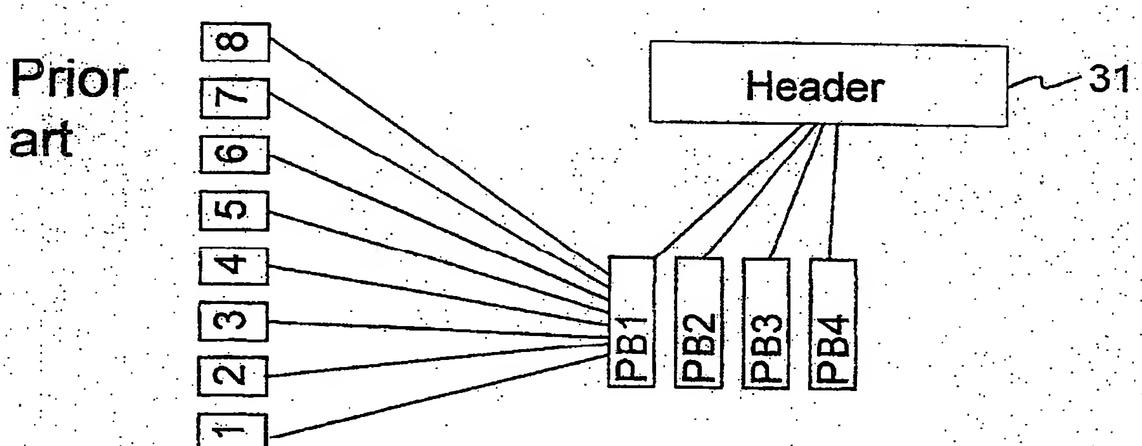

Prior art
8
7
6
5
4
3
2
1
Header
31
PB1
PB2
PB3
PB4

Fig. 5

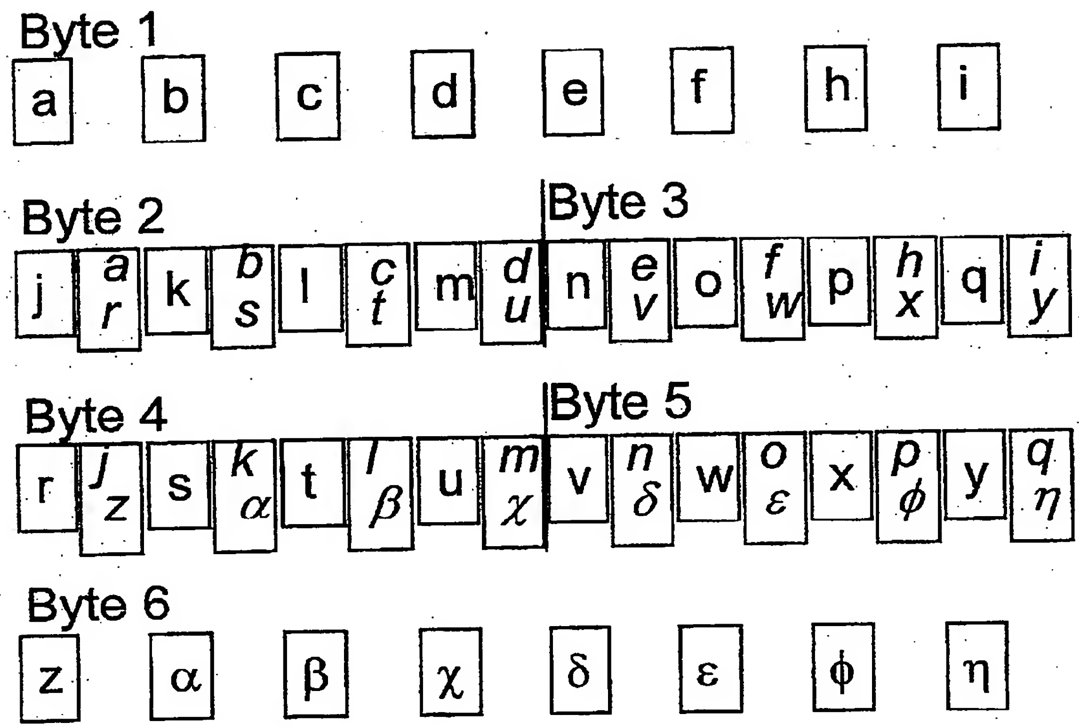

Byte 1
a b c d e f h i
Byte 2
Byte 3
j a r k b s l c t m d u n e v o f w p h x q i y
Byte 4
Byte 5
r j z s k α t l β u m χ v n δ w o ε x p φ y q η
Byte 6
z α β χ δ ε φ η

Fig. 6

S0: a, b, c, d, e, f, g, h
1, 1, 1, 1, 1,1,1, 1

S1: a, XOR(a,b), b, XOR(b,c), c, XOR(c,d), d,

XOR(d,e), e, XOR(e,f), f, XOR(f,g), g, XOR(g,h), h
= 1,0,1,0,1,0,1,0,1,0,1,0,1,0,1

S2: 1,0,1,0,**0**,0,1,0,1,0,1,0,1,**1**,1

S3: 1, 1, 1, 1, 1, 1, 1, 1

L1:	2	2	2	2	3	3	2	1
L0:	0	1	1	0	0	0	1	1

Fig. 7

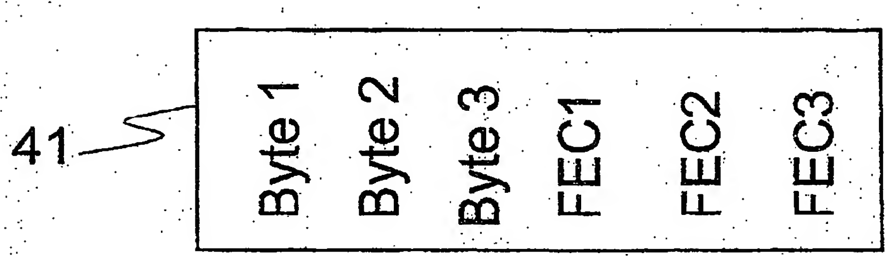


Fig. 8

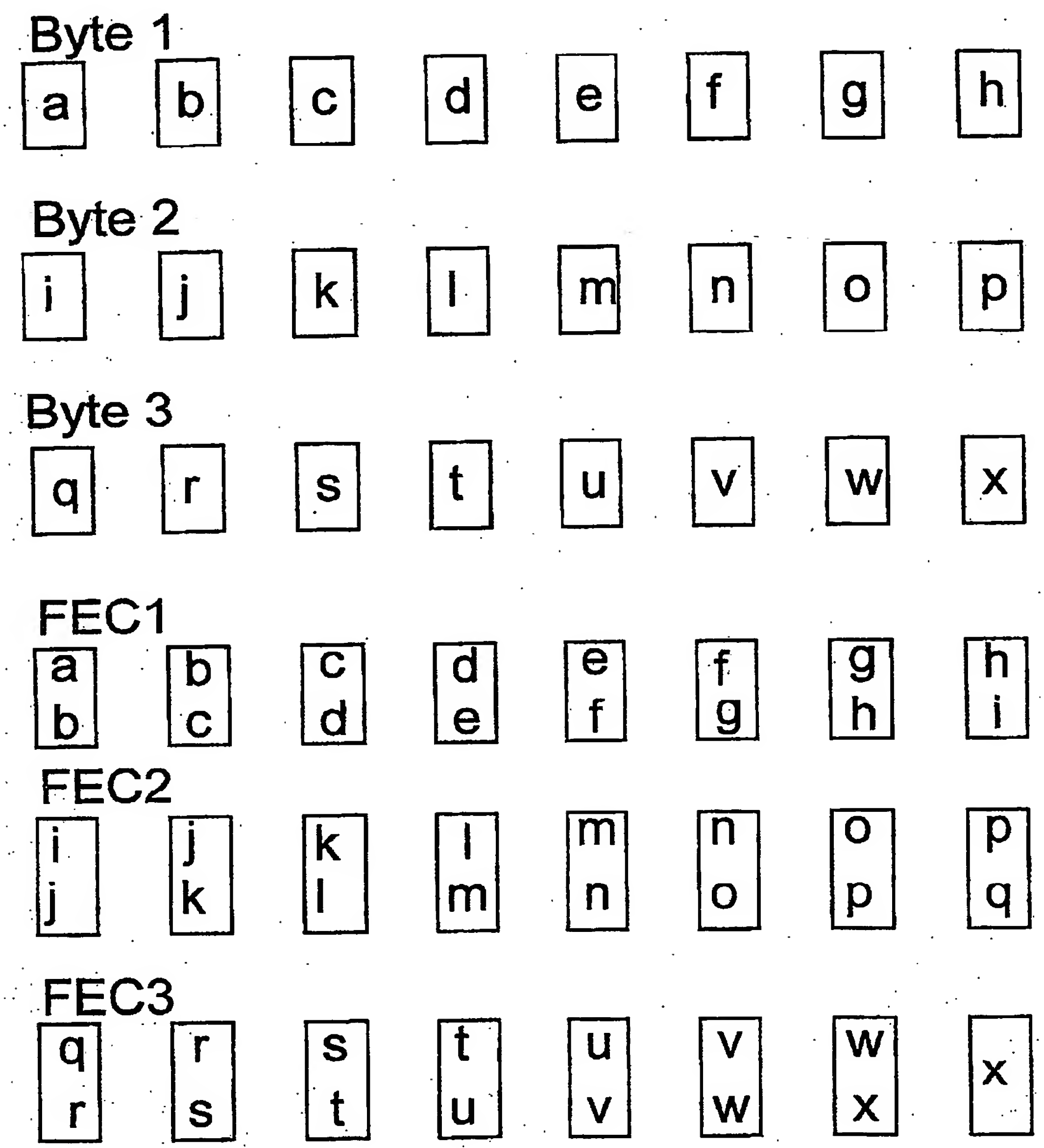

Byte 1
a b c d e f g h
Byte 2
i j k l m n o p
Byte 3
q r s t u v w x
FEC1
a b
b c
c d
d e
e f
f g
g h
h i
FEC2
i j
j k
k l
l m
m n
n o
o p
p q
FEC3
q r
r s
s t
t u
u v
v w
w x
x

Fig. 9

INTERNATIONAL SEARCH REPORT

International Application No
PCT, I 00/00233

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L1/00 H04L29/06

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L H03M H04Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MAMAIS G ET AL: "Evaluation of the Casner-Jacobson Algorithm for Compressing the RTP/UDP/IP Headers" THIRD IEEE SYMPOSIUM OF COMPUTERS AND COMM. 1998. ISCC '98. PROCEEDINGS, 1998, pages 543-548, XP002901178 Section I, II, IVB, V abstract ---	1-39
A	US 5 870 412 A (SCHUSTER G M ET AL) 9 February 1999 (1999-02-09) column 2, line 6 - line 27 ---	1-39

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
9 August 2000	22.09.2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	P. Gjervaldsaeter

1

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JACOBSON V: "Compressing IPUDPRTP Headers for Low-Speed Serial Links" S. CASNER CISCO SYSTEMS, [Online] February 1999 (1999-02), pages 1-20, XP002901179 Retrieved from the Internet: <URL:www.cis.ohio-state.edu/htbin/rtc2508.html> [retrieved on 2000-08-08] Section 3.1; 3.3.5 ---	1-39
A	JACOBSON V: "Compressing TCP/IP Headers for Low-Speed Serial Links" LBL, 1990, [Online] pages 1-29, XP002901180 Retrieved from the Internet: <URL:http://194.52.182.96/rfc/rfc1144.html> [retrieved on 2000-08-08] Section 3.2.2; 4 ---	1-39
A	US 5 309 450 A (KIM GYE-JONG) 3 May 1994 (1994-05-03) column 2, line 23 - line 62 ---	1-39
A	DRAVIDA S ET AL: "Error performance of IEEE 802.6 metropolitan area networks" INFOCOM '90, NINTH ANNUAL JOINT CONF. OF THE IEEE COMP. AND COMM. SOC. THE MULTIPLE FACETS OF INTEGRATION. PROCEEDINGS, IEEE, 1990, vol. 1, pages 105-115, XP002901181 Section II, IV -----	1-39

1

INTERNATIONAL SEARCH REPORT

In   ation on patent family members

International Application No
PCT. I 00/00233

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5870412 A	09-02-1999	AU 3062899 A	28-06-1999
		WO 9930462 A	17-06-1999
US 5309450 A	03-05-1994	KR 9501439 B	24-02-1995
		JP 7015352 A	17-01-1995

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☑ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☑ FADED TEXT OR DRAWING
- ☑ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☑ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: ______________________

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.